Mécanismes légers de gestion de la confiance pour des réseaux de capteurs sans fil

L. Vercouter[a] laurent.vercouter@emse.fr
J.-P. Jamont[b] jean-paul.jamont@iut-valence.fr
A. Balanel[a] anca.balanel@gmail.com

[a]Institut Henri Fayol,
École des Mines de Saint-Étienne, France

[b]Laboratoire de Conception et d'Intégration des Systèmes,
Université de Grenoble, France

Résumé

La communication en réseaux ad hoc, tels que des réseaux de capteurs sans fil, nécessite la mise en œuvre d'algorithmes décentralisés de routage impliquant que les nœuds du réseau adoptent un comportement précis. Le fonctionnement global du réseau dépend alors du bon comportement local des agents. Il en devient vulnérable à des comportements locaux déviants pouvant être causés par des pannes ou par l'intrusion d'agents malveillants. Nous proposons dans cet article une adaptation d'un protocole de routage pour réseaux de capteurs sans fil, le modèle MWAC, qui introduit un mécanisme décentralisé de gestion de la confiance de manière à détecter et éviter les noeuds déviants. L'algorithme de gestion de la confiance proposé suit une approche originale du fait qu'il est adapté à des agents déployés sur des infrastructures aux ressources limitées en énergie, mémoire et capacité de communication, caractéristiques des réseaux de capteurs sans fil. Il est notamment utilisable dans des systèmes multi-agents ne pouvant bénéficier d'infrastructures d'authentification et dans lesquels l'identité des agents est incertaine.

Mots-clés : *réseaux de capteurs sans fil, gestion de la confiance, protocole de routage sécurisé*

Abstract

Communication in ad hoc networks, such as Wireless Sensor Networks (WSN), requires the implementation of decentralized algorithms to manage routing. The global issue of such algorithms rely strongly on the correct behavior of nodes. This increases the vulnerability of the system regarding malicious intrusions or local failures. We propose in this article an adaptation of a routing protocol for WSN, the MWAC model, by introducing decentralized trust mechanisms in order to detect and avoid deviant nodes. The trust management algorithms follow an original approach as they are suited to a context of limited ressources in energy, memory and communication. They are thus suited to be used in multi-agent systems where authentication is not possible, and in which the agents' identity is uncertain.

Keywords: *wireless sensor networks, trust management, secure routing protocol*

1 Introduction

Les réseaux de capteurs sans fil fournissent une solution de collecte distribuée de données dans des environnements physiques parfois difficiles d'accès. Ils reposent généralement sur un très grand nombre de capteurs dispersés dans un espace appelé champ de captage. L'infrastructure matérielle des capteurs y est très légère afin d'en limiter son coût financier et sa consommation énergétique. Elle impose de fortes contraintes en ressources énergétiques et en mémoire. L'objectif du réseau étant de collecter les données des capteurs, chacun est équipé de capacités de communication pour transmettre les mesures captées à tout nœud dans son aire d'émission. La communication est non filaire et se fait de proche en proche (on parle de communications directes). Afin de pouvoir acheminer les mesures jusqu'à la station de collecte, il faut établir une communication d'un bout à l'autre. Il est alors nécessaire d'implémenter un algorithme de routage pour des réseaux *ad hoc* tenant compte des contraintes matérielles et d'une éventuelle dynamicité si les capteurs sont mobiles.

Parmi les algorithmes de routage proposés, nous nous intéressons ici plus particulièrement au modèle MWAC [5] qui propose une gestion du routage par un système multi-agent auto-organisé. L'approche auto-organisationnelle de ce protocole est particulièrement adapté au

contexte de réseaux de capteurs sans fil car elle permet de modifier dynamiquement l'organisation support du routage lorsque des perturbations du réseau apparaissent tel que des pannes de capteurs ou des déplacements dans l'espace. Comme tous les algorithmes décentralisés de gestion du routage, le protocole multi-saut employé et le maintien de tables locales de routage nécessitent que plusieurs nœuds du réseau exécutent une partie de l'algorithme. Si l'un de ces nœuds se comporte différemment cela met en péril le bon fonctionnement global du routage. MWAC, ainsi que la quasi-totalité des protocoles existants, est donc particulièrement vulnérable à l'exécution de codes incorrects sur un de ses nœuds, du fait de malveillances, de bogues ou de défaillances matérielles.

Nous proposons dans cet article une extension du modèle MWAC intégrant un mécanisme de gestion décentralisée de la confiance de manière à détecter et éviter les nœuds ne fonctionnant pas comme souhaité. Une version préliminaire de notre proposition [7] a consisté à introduire des algorithmes locaux de calcul de la confiance dans ses voisins permettant notamment de détecter et d'éviter tout nœud tentant d'usurper l'identité de la station de collecte. Les travaux présentés ici vont plus loin en étendant le calcul de la confiance à d'autres attaques sur le modèle MWAC visant à perturber la mise en place d'une auto-organisation efficiente. Une autre nouvelle contribution consiste à partager, entre nœuds voisins, la confiance qu'ils accordent à leur voisinage respectif pour accélérer la détection de nœuds malveillants.

Le problème de gestion de la confiance se pose de manière originale lorsque l'on considère des réseaux de capteurs. En effet, les fortes contraintes matérielles imposent la mise en place d'algorithmes très peu couteux. Il n'est notamment pas réaliste de supposer l'existence d'un système d'authentification (*e.g.* une infrastructure à clés publiques) permettant d'identifier chaque agent. C'est un verrou important à l'emploi de modèles existants pour la gestion décentralisée de la confiance dans un système multi-agent, car ils associent des valeurs de confiance à des agents clairement identifiés. Le modèle de confiance intégré à MWAC prend en compte cette spécificité et suit une approche originale d'estimation de la confiance adapté à l'absence d'authentification et à la facilité d'usurper une identité.

La seconde section décrit très brièvement le fonctionnement original de MWAC. La section 3 précise le problème posé sur la gestion de la confiance et pourquoi les approches existantes ne peuvent être employées, puis présente l'adaptation *TrustedMWAC* intégrant une estimation de la confiance dans les processus locaux de prise de décision. Enfin, des résultats expérimentaux obtenus en simulation sont présentés en section 4. La section 5 conclut cet article.

2 Le modèle MWAC

Un réseau de capteurs est constitué d'entités matérielles et logicielles qui sont chargées de tâches complexes et diverses : tâches d'acquisition de mesures, d'action, de comportement, de calcul, de communication. Ces entités peuvent interagir via des dispositifs de communication sans fil. La topologie est dynamique, la bande passante limitée et aucun organe dédié au routage n'est présent. Tous les nœuds du réseau ont une portée limitée et participent activement au routage de l'information (communication par sauts). Le routage est donc rendu plus difficile que dans les réseaux filaires traditionnels.

Les capteurs du réseau sont autonomes d'un point de vue énergétique. Un des objectifs globaux du système doit donc être de gérer au mieux les dépenses énergétiques. Quand il n'y a aucun envoi de message ces éléments sont généralement dans un mode de sommeil n'impliquant que de faibles dépenses d'énergie. Ce sont les phases de communication qui sont les plus coûteuses et qu'il convient d'optimiser par l'emploi d'un protocole de routage adapté. La solution idéale est d'avoir des routes optimales (généralement en termes de sauts) pour un coût d'obtention de la route aussi bas que possible. Dans le cas d'environnement agressif des fautes internes peuvent intervenir au niveau des constituants du réseau. Aussi l'infrastructure de communication doit être adaptative, tolérante aux pannes : un dysfonctionnement d'un des hôtes du réseau ne doit pas avoir un impact important sur le système et ne doit pas entraîner un coût énergétique d'adaptation élevé.

Le modèle MWAC [1] [5] a été conçu pour prendre en compte ces spécificités et gérer les communications dans les réseaux de capteurs.

1. *Multi Wireless Agent Communication*

2.1 L'auto-organisation dans MWAC

Le modèle MWAC organise les agents en groupes ayant une structure hiérarchique semblable à celle que l'on trouve dans les approches de type "clusters". L'intérêt de ce type d'approche est de localiser au mieux l'inondation inhérente aux réseaux de capteurs, c'est-à dire la diffusion de messages à tous les nœuds voisins, et ainsi d'obtenir un gain conséquent en énergie.

Les agents sont rassemblés en groupes constitués de :

- Un agent *représentant* qui administre les communications au sein de son groupe. C'est lui qui est chargé de trouver une route vers les destinataires des messages qu'envoient ses administrés.
- Un ou plusieurs agents de *liaison* : ils appartiennent à plusieurs groupes et permettent ainsi aux différents représentants de communiquer entre eux,
- Aucun ou plusieurs *simples membres* qui n'ont aucun rôle particulier dans le groupe si ce n'est recevoir et traiter les messages qui leurs sont destinés et transmettre leurs propres messages (ils ne participent pas à l'acheminement des messages).

La figure 1 donne un aperçu de la structure en groupes formée selon MWAC. Nous pouvons notamment observer que le nombre de liens inondés est beaucoup moins important lorsque le réseau est auto-organisé.

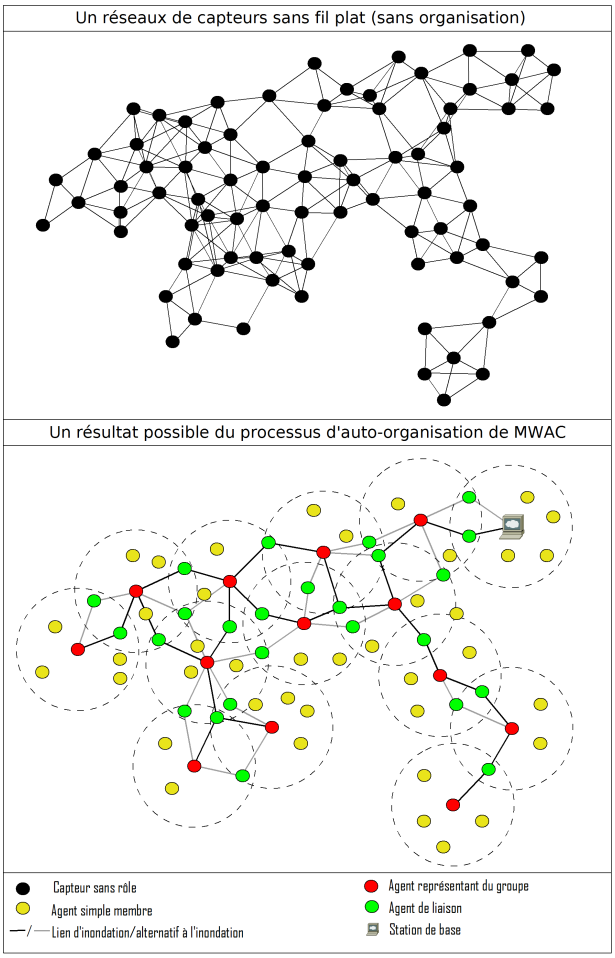

FIGURE 1 – Organisation dans MWAC

L'efficacité du routage dépend de l'allocation des rôles aux agents et du maintien d'une organisation cohérente. Les agents échangent périodiquement des messages d'introduction contenant : (i) leur *identifiant* ; (ii) leur *rôle* ; (iii) les *groupes* auxquels ils appartiennent. Un agent s'attribue localement un rôle en fonction du rôle perçu de ses voisins. L'idée générale est que si un agent n'a aucun représentant dans son voisinage, il créé un nouveau groupe en devenant lui-même représentant. S'il a un représentant dans son voisinage, il prend le rôle de simple membre. Si plusieurs représentants sont dans son voisinage, il joue le rôle de connection entre ces groupes.

Un conflit apparaît si deux représentants peuvent directement communiquer. Dans ce cas, un protocole de résolution de conflits leur permet de décider de l'agent qui abandonne son rôle de représentant en comparant un score calculé essentiellement sur le nombre de voisins et l'énergie disponible de chacun. Le détail des algorithmes de MWAC est disponible dans l'article de référence du modèle [5].

2.2 Vulnérabilité face aux pannes et intrusions

MWAC est particulièrement adapté pour gérer la communication dans des réseaux de large échelle, dotés de capteurs aux ressources limitées. Cependant, comme c'est le cas dans la plupart des algorithmes décentralisés, cela suppose que chaque agent se comporte comme attendu. Or un réseau de capteurs déployé est souvent ouvert dans le sens où l'on peut ajouter en cours d'exécution de nouveaux capteurs. Ceci facilite la possibilité d'intrusion d'agents malveillants dans l'optique de nuire au fonctionnement global du réseau. La fragilité des capteurs et les risques liés aux environnements de déploiement accroissent aussi le risque de défaillance.

L'adoption d'une approche auto-organisé per-

met à MWAC de s'adapter dynamiquement à certaines pannes. Par exemple, si un capteur "disparaît" par manque d'énergie, l'organisation s'adaptera en modifiant les rôles des agents de manière à conserver des routes valides. Par contre si la défaillance entraine l'envoi de messages dont le contenu est faux ou si ceux-ci sont volontairement falsifiés, le processus d'auto-organisation ne résoudra pas ce problème et sera même fortement perturbé par la définition de rôles et de groupes inadaptés.

Nous nous intéressons ici plus particulièrement aux mensonges survenant dans le message d'introduction. Un mensonge sur une identité permettra à un agent de récupérer les messages à destination d'un autre (par exemple la station de collecte). La section suivante propose une adaptation de ce modèle, nommée *TrustedMWAC*, qui introduit la notion de confiance dans son voisinage pour détecter des comportements déviants dans les voisins de chaque nœud.

3 L'extension *TrustedMWAC* pour une auto-organisation sécurisée

L'approche adoptée par de nombreux travaux existants (cf [6] pour en avoir une revue) pour réduire la vulnérabilité de systèmes décentralisés est d'introduire la notion de confiance. Il est cependant difficile d'utiliser ces solutions dans le cas de réseaux de capteurs sans fil, notamment à cause des limitations sur leur coût énergétique et de communication, imposées par la légèreté des supports matériels.

Après avoir justifié en quoi les approches existantes ne conviennent pas à des réseaux de capteurs sans fil, nous décrivons en seconde sous-section notre proposition d'un modèle léger de gestion de la confiance, adapté au protocole MWAC. La dernière partie de cette section présente une seconde extension permettant le partage d'informations sur la confiance dans le but d'améliorer la détection de comportements déviants.

3.1 Gestion décentralisée de la confiance pour des réseaux de capteurs sans fil

Les systèmes de gestion de la confiance ont pour objectif de protéger un système contre de mauvais comportements des entités qui le composent. L'idée générale de ces travaux est d'observer les comportements des agents, d'évaluer leur conformité par rapport à ce qui est attendu, de calculer et d'affecter en conséquence des valeurs de confiance aux autres agents, puis d'éviter toute interaction avec les agents jugés indignes de confiance. Cette dernière étape a pour effet d'exclure socialement des agents du système par un mécanisme dit de contrôle social [1]. Certains systèmes de gestion de la confiance ont déjà été proposés pour sécuriser le routage en fonctionnant de manière décentralisée, par exemple pour des réseaux *ad hoc* [3] ou des réseaux pair-à-pairs [2, 8].

Une hypothèse partagée par tous les modèles de gestion de la confiance en système multi-agents est que les agents soient authentifiés. Il est en effet indispensable qu'une identité soit attachée à chaque agent et qu'elle ne puisse être contestée de manière à associer identités et valeurs de confiance. Il est généralement admis que des services d'infrastructure, tel qu'une infrastructure à clés publiques, peuvent être utilisés. Dans le cas de réseaux de capteurs sans fil, cette hypothèse est problématique car la très faible capacité de stockage et de communication des capteurs n'autorise pas le déploiement de telles infrastructures. Il n'est pas réaliste de considérer que chaque nœud stocke les clés publiques de tous les autres capteurs, ni qu'ils soient capables à tout moment de contacter un serveur central les conservant.

L'absence d'authentification rend alors impossible l'usage de modèles classiques de confiance. En effet, quand un agent reçoit un message, celui-ci peut provenir de n'importe lequel de ces voisins présents dans sa portée de communication, revendiquant n'importe quelle identité. Une gestion décentralisée de la confiance pour des réseaux de capteurs pose ainsi un problème original nécessitant l'adoption d'une nouvelle approche.

L'approche suivie dans le modèle décrit ici consiste à utiliser la notion de confiance pour estimer la fiabilité d'un *voisinage* dans son ensemble plutôt qu'une évaluation indépendante de chaque voisin. Un voisinage indigne de confiance doit être interprété comme incluant au moins un agent malveillant ou défaillant. Lorsqu'un agent croit que son voisinage n'est pas digne de confiance, il adopte un fonctionnement *dégradé* n'effectuant que les tâches minimales attendues de lui. Ce fonctionnement dégradé doit éviter l'implication du voisinage de l'agent à toute tâche collective. Notre objectif est que tous les voisins d'un nœud déviant détectent progressivement l'existence de mauvais

Mensonge sur	**No**	**Détection par le nœud n**	**Baisse de la confiance**
id	1	si id utilisé = id de n	$Trust(id)$ = 0
	2	si id utilisé = id de la station de collecte	voir plus bas le processus *new group checking*
$group$	3	si n est le représentant d'un groupe et que son groupe n'est pas inclus dans l'ensemble des groupes du message	$Trust(id)$ = 0
	4	si un nouveau groupe G est présenté et que l'agent qui l'introduit est le seul agent à faire une liaison avec ce groupe.	voir plus bas le processus *new group checking*
$role$	5	si le nœud revendique un rôle de liaison vers un groupe G.	voir plus bas le processus *new group checking*

TABLE 1 – Mensonges dans le message d'introduction

comportements et passent en mode dégradé. La partie du réseau composé du nœud déviant et de ses voisins se retrouvera alors mis en *quarantaine* sans possibilité d'influencer le fonctionnement global du système. Si une écoute flottante est possible (dans une communication sans fil, il n'est pas irréaliste de supposer qu'un nœud peut ainsi intercepter des messages ne lui étant pas destinés), un nœud pourra utiliser tous les messages émis dans son voisinage, et pas seulement ceux qui lui sont adressés, pour détecter d'éventuels mauvais comportements.

3.2 Estimation de la confiance

Chaque agent réalise localement des estimations de confiance en observant les messages émis dans son voisinage. Même si l'authentification est impossible, les agents doivent utiliser un identifiant (véritable ou usurpé) lors de l'envoi d'un message. Nous proposons ici d'estimer la confiance qu'à un agent dans l'*usage d'un identifiant* (plutôt que dans un agent authentifié par une identité).

Initialisation et mise à jour. Une nouvelle valeur de confiance est créée à chaque fois qu'un nouvel identifiant id est utilisé dans le voisinage d'un agent. Cette valeur est comprise dans l'intervalle $[0; 1]$ avec pour valeur initiale une confiance maximale ($Trust(id) = 1$).

La confiance est calculée et mise à jour par une analyse des messages envoyés par un agent utilisant un identifiant donné. Si le message est jugé incorrect par rapport à un comportement communicatif attendu nous parlerons de *mensonge*. La confiance dans un identifiant est baissée lorsqu'un mensonge l'utilisant est détecté.

Détection de mensonge. Selon les cas d'observation, un mensonge peut être reconnu de manière sûre par un agent ou seulement suspecté. Dans le premier cas, la réduction doit être drastique car l'agent est assuré que son voisinage contient un nœud déviant. Dans le second cas, un événement inhabituel peut révéler un mensonge mais il existe quand même des cas exceptionnels où un tel message peut apparaître dans un fonctionnement normal. Par exemple, un nœud peut envoyer une information fausse si sa croyance locale n'est pas à jour. Dans le cas de MWAC, cela peut se produire si un nœud affirme appartenir à un groupe alors qu'il vient de sortir du champ de communication du représentant du groupe mais qu'il ne l'a pas encore détecté. Si un mensonge est seulement suspecté, la confiance sera baissée avec une importance proportionnelle à la probabilité qu'une déviance en soit la cause. Le tableau 1 présente les mensonges possibles dans MWAC, la manière de les détecter et quelle réaction le nœud qui les détecte doit adopter.

Les situations no1 et no3 font respectivement référence au cas où un nœud perçoit un message utilisant son propre identifiant et au cas où un représentant perçoit un message venant d'un de ses voisins n'indiquant pas qu'il appartient au groupe du représentant. Dans ces cas, le message est indéniablement un mensonge visant soit à usurper l'identité d'un nœud, soit à cacher l'existence d'un groupe. La confiance est alors abaissée au plus bas niveau.

La situation no2 correspond au cas où la station de collecte, supposée fixe apparaît à côté d'un nœud pendant son exécution. Cela reste possible car le capteur peut être mobile mais il se peut aussi qu'un agent tente d'usurper l'identité de la station de collecte. Les quatrième et cinquième situations où un voisin déclare appartenir à un nouveau groupe et éventuellement de-

venir nœud de liaison peuvent aussi arriver du fait d'une mobilité. Cela peut aussi être une tentative d'altérer le routage. Dans ces trois cas, le mensonge est ici suspecté mais sans certitude.

La réaction à cette suspicion consiste alors à démarrer un processus (*new group checking*) visant à envoyer une requête à la station de collecte (cas no2) ou au représentant du groupe G (cas no4 et no5) lui demandant si le nœud v est bien dans son voisinage. Cette requête est envoyée par inondation à tous les voisins du nœud suspectant le mensonge et précise que la route de ce message doit éviter le nœud v. À chaque fois que le destinataire final reçoit cette requête, il y répond en indiquant si le nœud suspect v est dans son voisinage ou non. Il est bien entendu possible que v puisse percevoir ce message, car il est dans le voisinage du nœud l'ayant initialement émis et renvoie une fausse réponse usurpant encore l'identifiant concerné. Suivant les réponses reçues à cette requête, la confiance est mise à jour comme indiqué dans le tableau 2.

Réponses reçues	**Baisse de la confiance**
Aucune réponse	pas de baisse
Toutes les réponses indiquent que v est dans le groupe G	pas de baisse
Toutes les réponses indiquent que v n'est pas dans le groupe G	$Trust(id) = Trust(id) - \alpha$
Des réponses différentes sont reçues	$Trust(id) = Trust(id) - \beta$

TABLE 2 – Issue du processus *new group checking process*

Si aucune réponse n'est reçue ou si toutes indiquent que v est dans le groupe G, il n'y a probablement pas de mensonge. Si toutes les réponses indiquent que v n'est pas dans le groupe G, il peut soit y avoir un mensonge soit une croyance de v temporairement fausse. La confiance est alors réduite d'une valeur α mais pas à la valeur minimale.

Le dernier cas consiste en une situation où des réponses différentes arrivent. Il se peut que celà soit dû à de fausses réponses renvoyées et que v soit un usurpateur. Il reste néanmoins une possibilité plus rare dans laquelle v est dans le groupe G quand son représentant reçoit et répond à une partie des requêtes et l'ait quitté quand il répond aux requêtes restantes. La sanction β sur la confiance doit ici être plus importante mais pas encore maximale. Nous proposons de fixer ces sanctions dans l'intervalle $0 < \alpha < \beta < 1$. La valeur minimale de la confiance est fixée à 0 et ramenée à cette valeur si une décroissance amène une confiance négative.

Recouvrement de la confiance. Il est nécessaire que la confiance dans un voisinage puisse être rétablie. La mobilité, même faible, du réseau fait qu'un nœud malveillant peut être amené à quitter un voisinage. Il peut aussi être tout simplement retiré du réseau ou ne plus fonctionner faute d'énergie. Il se peut également qu'une défiance vis-à-vis d'un voisinage ne soit pas la cause d'une malveillance mais d'une co-occurrence d'événements exceptionnels ayant entrainés plusieurs suspicions. Le rétablissement de la confiance s'opère ici par un phénomène d'oubli avec une lente augmentation de la confiance avec le temps. L'algorithme 1 implémente ce phénomène.

Algorithm 1 Algorithme de recouvrement de la confiance

for all id in $neighborhood.getUsedIds()$ **do**
$Trust(id) = (1 - \lambda) * Trust(id) + \lambda$
end for

Deux facteurs paramètrent la vitesse du rétablissement de la confiance : (i) la fréquence ν à laquelle l'algorithme de rétablissement est invoqué ; (ii) le taux d'évaporation de la sanction λ, avec $0 \leq \lambda < 1$ indiquant la quantité de confiance regagnée à chaque cycle. La valeur de ces paramètres dépend principalement de la mobilité supposée des nœuds. En présence d'une mobilité forte, nous recommandons une fréquence et un taux d'évaporation importants car les voisinages devraient souvent varier. Si la mobilité est faible, le réseau sera plutôt statique et il est préférable de ralentir le rétablissement avec des valeurs plus faibles pour ces paramètres.

Prise de décision de confiance. La confiance dans les identifiants est utilisée pour estimer la confiance du voisinage dans son ensemble. Le voisinage d'un nœud est jugé indigne de confiance s'il existe au moins un identifiant dans lequel le nœud n'a pas confiance.

Le seuil θ représente la valeur de confiance minimale en dessous de laquelle un nœud est jugé indigne de confiance. Ce seuil prend une valeur dans $0 < \theta < 1$.

Algorithm 2 Algorithme de décision de confiance dans le voisinage

for all id in $neighborhood.getUsedIds()$ **do**
  if $Trust(id) < \theta$ **then**
    return $distrusted$
  end if
end for
return $trusted$

3.3 Adaptation du modèle MWAC

Nous avons adapté le modèle original MWAC pour y intégrer la notion de confiance dans un voisinage. Lorsqu'un nœud n'a pas confiance dans son voisinage, il adopte un fonctionnement dégradé et participe au minimum au routage. Le mode dégradé correspond dans MWAC à un nouveau rôle *BACKUP*. L'algorithme 3 est une adaptation du processus d'affectation des rôles.

Algorithm 3 Adaptation de l'affectation des rôles

if $neighborhood.isEmpty()$ **then**
  // No possible organizational structure
else if $neighborhood.trust() = distrusted$ **then**
  $myRole \leftarrow BACKUP$
else if $myRole = REPRESENTATIVE$ and $neighborhood.nbOfRepresentative() > 0$ **then**
  $conflictResolutionProcedure()$
else if $neighborhood.nbOfRepresentative() = 0$ **then**
  $myRole \leftarrow REPRESENTATIVE$
else if $neighborhood.nbOfRepresentative() = 1$ **then**
  $myRole \leftarrow SIMPLEMEMBER$
else $\{neighborhood.nbOfRepresentative() > 1\}$
  $myRole \leftarrow CONNECTION$
end if

Le nouveau rôle *BACKUP* correspond au même fonctionnement que le rôle de simple membre, excepté qu'il est impossible d'évoluer vers un rôle de représentant ou de liaison.

3.4 Mise en quarantaine

L'échange de recommandations ou/et la construction collective de réputations est une pratique classique en gestion de la confiance pour augmenter le nombre d'informations prises en entrée et accélérer l'apprentissage de valeurs de confiance précises. Cependant, l'absence d'authentification nous empêche l'usage de ces techniques car elles nécessitent aussi d'associer une valeur de confiance ou de réputation à une identité indéniable.

L'échange de valeur de confiance ne présente dans notre cas que peu d'intérêt. Par contre, le fait de savoir que certains voisins d'un nœud ont adopté un rôle $BACKUP$ est utile. Cela signifie qu'il y a probablement un agent malveillant à proximité et peut-être dans le voisinage direct du nœud qui reçoit cette information.

Afin de partager cette information, nous proposons ici une seconde variante de MWAC où les agents prennent en compte l'éventuel rôle $BACKUP$ de leurs voisins. Un nœud informe ses voisins de son rôle dans le message classique d'introduction envoyé périodiquement, et dans le cas du rôle $BACKUP[id*]$, il y ajoute les identifiants ($[id*]$) en lesquels il n'a pas confiance. L'objectif est de propager ce message à tous les voisins présumés du nœud ayant cet identifiant afin que ceux-ci passent également en mode dégradé. Si tout le voisinage du nœud déviant est en mode dégradé, son entourage dans le réseau sera mis en quarantaine et il ne pourra plus nuire au bon fonctionnement du système.

La prise en compte du rôle des voisins afin de partager les informations de confiance nécessite une adaptation de l'algorithme (no2) de décision de confiance telle que présentée par l'algorithme 4.

Algorithm 4 Décision de confiance intégrant le rôle des voisins

for all id in $neighborhood.getUsedIds()$ **do**
  if $Trust(id) < \theta$ **then**
    return $distrusted$
  end if
  if $(role(id) = BACKUP(IDS))$ and $(IDS \cap neighborhood.getUsedIds() \neq \emptyset)$ **then**
    return $distrusted$
  end if
end for
return $trusted$

Il est à noter que, pour éviter que la suspicion se propage à tout le réseau, les identifiants ayant provoqué le passage dans le rôle $BACKUP$ sont communiqués. Ainsi seuls les nœuds ayant perçu un de ses identifiants dans leur voisinage passent en mode dégradé. La propagation s'arrête dès que l'on s'éloigne de ces identifiants.

4 Évaluation expérimentale

L'apport de l'introduction de la confiance pour la robustesse du protocole MWAC a été évalué expérimentalement sur le simulateur MASH [4].

Cette section décrit la configuration des expériences menées et les résultats obtenus face à l'introduction d'agents malveillants.

4.1 Protocole expérimental

L'instrumentation de réseaux de capteurs sans fil fait généralement intervenir deux types de nœuds : les capteurs et une station de collecte. Les capteurs servent de dispositif distribué d'acquisition de données alors que la station de collecte doit récupérer toutes ces données pour les traiter. Nous considérons ici des réseaux ne contenant qu'une seule station de collecte comme c'est traditionnellement le cas.

Une attaque contre le bon fonctionnement du réseau consiste à tenter d'empêcher des données d'atteindre la station de collecte. Un nœud malveillant peut pour cela adopter différentes stratégies. La plus efficace consiste à se faire passer pour la station de collecte de manière à attirer les messages sans les transmettre.

Les expériences décrites ici se sont déroulées sur des réseaux composés de 100 capteurs. Le temps est discrétisé et chaque simulation se déroule pendant 100 pas de temps. À chaque pas de temps, chaque capteur prend une mesure et émet un message à destination de la station de collecte. Pendant une période d'initialisation, la simulation se déroule avec un comportement correct pour tous les nœuds. Cela permet la mise en place d'une première auto-organisation et correspond au déploiement initial du réseau lors duquel on peut supposer que tous les capteurs sont fonctionnels et qu'il n'y a pas encore eu d'intrusion malveillante. Au pas de temps no20, un dysfonctionnement est introduit et l'impact de cet événement sur le routage des messages est analysé.

Les simulations ont été effectuées par une adaptation du code de MWAC implémenté dans le simulateur MASH [4]. La figure 2 montre une copie d'écran tirée du simulateur faisant apparaître la topologie du réseau en fond et le détail d'un nœud dans une boîte de dialogue. Celle-ci donne un exemple de la confiance qu'a le nœud observé dans les identifiants 48 et 50. Sa faible confiance dans l'identifiant 48 implique une défiance dans son voisinage.

Pour ces simulations, les paramètres du modèles ont été fixés à : $\theta = 0.5; \alpha = 0.4; \beta = 0.8$ et $\lambda = 0$

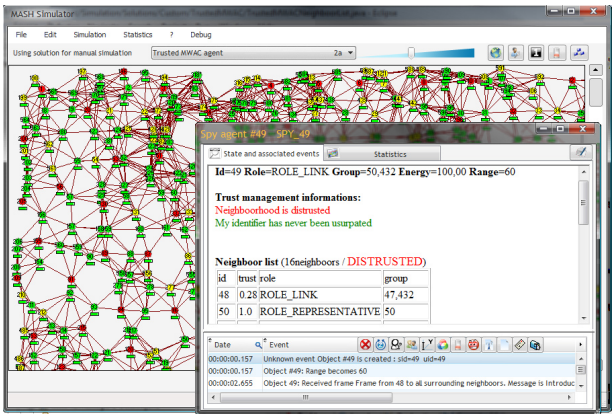

FIGURE 2 – Le simulateur MASH

4.2 Scénario d'usurpation de l'identité de la station de collecte

Un premier scénario simule l'introduction de capteurs usurpant l'identité de la station de collecte. Le capteur malveillant déclare à son voisinage être la station de travail dans le but d'attirer les messages contenant les mesures et d'empêcher la réelle station de collecte de les recevoir. La figure 3 présente la perte de messages subie par la station de collecte après usurpation de son identité.

Un premier test a été réalisé sans usurpation pour vérifier que 100% des messages étaient bien reçus, ce qui est le cas. Avec la version originale de MWAC, l'usurpation d'identité de la station de collecte a un impact important. Lorsqu'un usurpateur apparaît, près de 30% des mesures sont perdues après 100 pas de temps. Une autre simulation a été réalisée avec 2 usurpateurs et la perte y est d'environ 42%. Les nœuds représentant vont en effet créer de nouvelles tables de routage vers ce qu'ils croient être la station de collecte et choisir les routes les plus courtes. Ceux qui sont plus proches de la station de collecte continuent à bien lui envoyer leurs mesures alors que ceux plus proches d'un usurpateur lui enverront. Le nombre d'usurpateurs et leur position dans le réseau a ainsi une influence sur le volume de mesures perdues.

Les deux courbes intitulées TrustedMWAC montre l'apport des algorithmes de confiance pour limiter l'impact des usurpateurs. Les simulations ont été effectuées dans la même topologie et configuration de réseau que pour MWAC. Il y a beaucoup moins de mesures perdues, environ 3% et 6% pour respectivement un et deux usurpateurs. Les nœuds voisins d'un

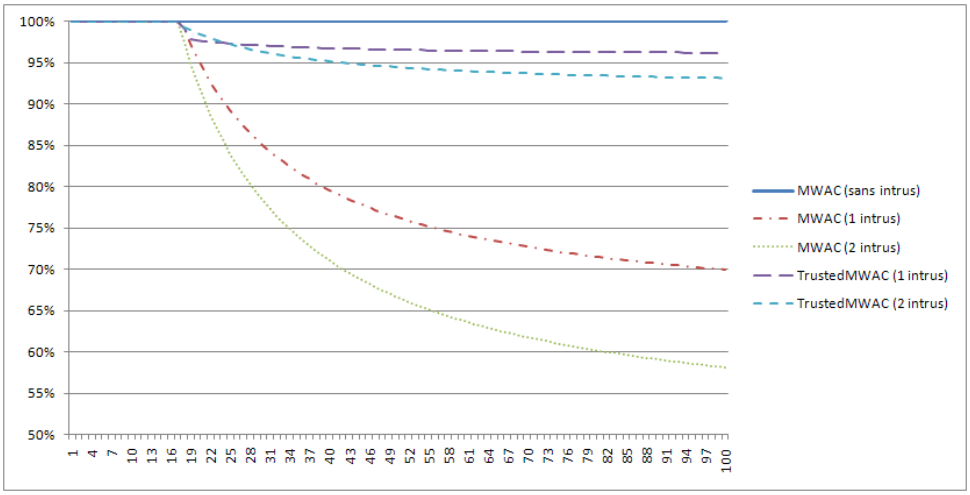

FIGURE 3 – Usurpation de l'identité de la station de collecte (% de messages reçus après une attaque)

usurpateur vont en effet demander par toutes leurs routes vers la station de collecte si elle se situe bien dans leur voisinage et, après réception de réponses contradictoires, passer en mode dégradé. Les mesures qui restent perdues sont celles des capteurs usurpateurs et d'éventuels autres nœuds isolés du réseau. L'usurpateur continuant de clamer une fausse identité, la confiance ne sera pas recouvrée

4.3 Scénario d'usurpation d'identité

Dans le second scénario simulé, l'attaque est une usurpation d'identité d'un capteur voisin. Le nœud déviant va envoyer des messages en prenant l'identité d'un de ses voisins jouant le rôle de représentant. Pour les nœuds voisins de ce représentant et de l'usurpateur, il n'y aura aucun changement car ils n'ont pas la possibilité de distinguer l'émetteur réel du message. Pour ceux qui ne sont pas dans le voisinage du représentant réel, l'apparition d'un capteur ayant cette identité peut résulter d'une simple mobilité. Le seul agent capable de détecter l'usurpation est le représentant dont l'identité a été volée. La figure 4 donne les résultats de l'expérimentation de ce scénario.

L'impact de cette attaque en utilisant MWAC est plus faible que dans le cas précédent, entre 6% et 7% des mesures sont perdues. Il s'agit des mesures ayant transitées par l'usurpateur et que le représentant réel n'a pas perçues (sinon il les fait suivre normalement par une route vers la station de collecte). L'usage de TrustedMWAC permet au représentant réel de détecter l'usurpation et de passer en mode dégradé. Cependant si la mise en quarantaine présentée en section 3.4 n'est pas effectuée, il y a plus de mesures perdues (environ 11%). En effet, le réel représentant, en passant en mode dégradé ne joue plus son rôle et laisse l'usurpateur traiter tous les messages à son intention.

La prise en compte du rôle des voisins pour créer une zone de quarantaine donne de bien meilleurs résultats. Environ 2% des mesures y sont perdues, correspondant aux nœuds isolés par la quarantaine. Dans ce cas, tous les voisins de l'usurpateur et du représentant réel (car il est impossible de distinguer les deux) passent en mode dégradé pour isoler ces capteurs.

5 Conclusion

Le routage des informations en réseau de capteurs sans fil repose sur la mise en place d'un réseau *ad hoc* entre les capteurs et d'un algorithme décentralisé de routage. Le modèle MWAC [5] propose de déployer des agents embarqués sur les capteurs qui s'auto-organisent de manière à mettre en place dynamiquement une structure organisationnelle facilitant le routage. C'est une solution légère adaptée au déploiement sur des capteurs aux ressources matérielles limitées. Elle reste néanmoins vulnérable à une défaillance, intentionnelle ou non, du fonctionnement local des agents.

Cet article propose d'intégrer un modèle de calcul et de décision de la confiance à MWAC dans une extension nommée TrustedMWAC. Les contraintes fortes sur le coût des algorithmes employés interdisant l'emploi de sys-

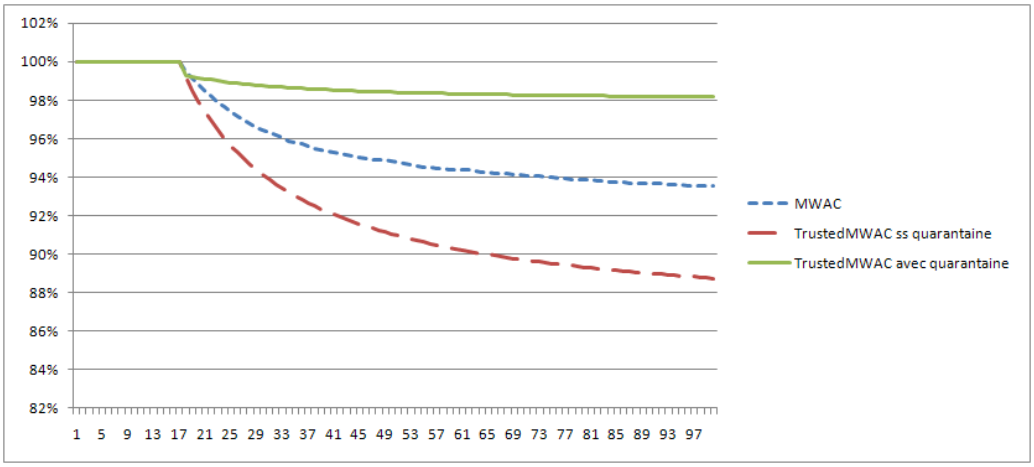

FIGURE 4 – Usurpation de l'identité d'un voisin (% de messages reçus après une attaque)

tèmes classiques de gestion de la confiance, TrustedMWAC utilise un mécanisme de gestion de la confiance très léger, fonctionnant notamment en l'absence d'un système d'authentification. L'approche suivie est que chaque agent évalue la confiance qu'il a dans son voisinage, plutôt que séparément dans chaque voisin. En cas de défiance, il se sacrifie en adoptant un fonctionnement dégradé. Lorsque tous les nœuds voisins d'un agent défaillant sont en mode dégradé, il se retrouve dans l'incapacité de nuire globalement au système. Le secteur qu'il compose avec ses voisins est mis en quarantaine. Des expérimentations réalisés sur une adaptation du simulateur de MWAC montrent l'efficacité de cette approche.

Le travail présenté ici s'est focalisé sur les mécanismes d'auto-organisation de MWAC, proposant un rôle spécifique au mode dégradé rendant l'auto-organisation plus robustes. Afin de développer une méthode globalement robuste, nos travaux s'orientent maintenant vers l'usage de la confiance pendant le routage, c'est-à-dire la vérification qu'un agent, dans une organisation donnée, se comporte bien tel que son rôle lui dicte. Une évaluation du déploiement sur un réseau réel de capteurs est également prévue.

Références

[1] Cristiano Castelfranchi. Engineering social order. In *Proceedings of the 2nd International Workshop on Engineering Societies in the Agent's World (ESAW'00)*, 2000.

[2] Zoran Despotovic and Karl Aberer. P2P reputation management : Probabilistic estimation vs. social networks. *Journal of Computer Networks, Special Issue on Management in Peer-to-Peer Systems : Trust, Reputation and Security*, 50(4) :485–500, 2006.

[3] Nathan Griffiths, Arshad Jhumka, Anthony Dawson, and Richard Myers. A simple trust model for on-demand routing in mobile ad-hoc networks. In Costin Badica, Giuseppe Mangioni, Vincenza Carchiolo, and Dumitru Dan Burdescu, editors, *Proceedings of the 2nd International Symposium on Intelligent Distributed Computing - IDC 2008*, volume 162 of *Studies in Computational Intelligence*, pages 105–114, Catania, Italy, 2008. Springer.

[4] Jean-Paul Jamont and Michel Occello. A multiagent tool to simulate hybrid real/virtual embedded agent societies. In *IAT*, pages 501–504, 2009.

[5] Jean-Paul Jamont, Michel Occello, and André Lagrèze. A multiagent approach to manage communication in wireless instrumentation systems. *Measurement*, 43(4) :489 – 503, 2010.

[6] Jordi Sabater and Carles Sierra. Review on computational trust and reputation models. *Artif. Intell. Rev.*, 24(1) :33–60, 2005.

[7] Laurent Vercouter and Jean-Paul Jamont. Lightweight trusted routing for wireless sensor networks. In *9th Conference on Practical Application of Multi-Agent Systems (PAAMS'11)*, Salamanca, 2011.

[8] Laurent Vercouter and Guillaume Muller. L.I.A.R. : achieving social control in open and decentralized multiagent systems. *Applied Artificial Intelligence*, 24 :723–768, September 2010.